CLAIMS

What is claimed is:

1. A logic circuit comprising:

a control flow logic to select a trace descriptor for processing including at least one dependency descriptor including dependency information for each instruction sequence; and

a data flow logic coupled to the control flow logic to execute a plurality of instruction sequences according to the dependency information stored in the dependency descriptor.

2. The logic circuit of claim 1 further comprising a first storage area coupled to the control flow logic and the data flow logic, the first storage area to store the dependency descriptor.

3. The logic circuit of claim 2 further comprising a second storage area coupled to the control flow logic, the second storage area to store a trace descriptor.

4. The logic circuit of claim 3 further comprising a third storage area coupled to the data flow logic, the third storage area to store instructions contiguously based on dependency information.

5. The logic circuit of claim 4 further comprising a fourth storage area coupled to the data flow logic and control flow logic, the fourth storage area to store live-out data.

6. The logic circuit of claim 5 further comprising a fifth storage area coupled to the control flow logic, the fifth storage area to map live-in and live-out data.

7. The logic circuit of claim 6 wherein each of the storage areas are in at least one memory device.

8. The logic circuit of claim 1 wherein the trace descriptor includes aggregate live-in data for the at least one dependency descriptor.

9. The logic circuit of claim 1 wherein the trace descriptor includes aggregate live-out data for the at least one dependency descriptor.

10. A computer system comprising:

at least one memory device;

a bus coupled to the at least one memory device;

a control flow logic to analyze dependencies among instruction sequences and creates a dependency descriptor including dependency information for each instruction sequence; and

a data flow logic coupled to the control flow logic to execute a plurality of the instruction sequences according to the dependency information stored in the dependency descriptor.

11. The computer system of claim 10 further comprising a first storage area coupled to the control flow logic and the data flow logic, the first storage area to store the dependency descriptor.

12. The computer system of claim 11 further comprising a second storage area coupled to the control flow logic, the second storage area to store a trace descriptor.

13. The computer system of claim 12 further comprising a third storage area coupled to the data flow logic, the third storage area to store instructions contiguously based on dependency information.

14. The computer system of claim 13 further comprising a fourth storage area coupled to the data flow logic and control flow logic, the fourth storage area to store live-out data.

15. The computer system of claim 14 further comprising a fifth storage area coupled to the control flow logic, the fifth storage area to map live-in and live-out data.

16. The computer system of claim 15 wherein each of the storage areas are in at least one memory device.

17. The computer system of claim 10 wherein the trace descriptor includes aggregate live-in data for the at least one dependency descriptor.

18. The computer system of claim 10 wherein the trace descriptor includes aggregate live-out data for the at least one dependency descriptor.

19. The computer system of claim 10 wherein the dependency descriptor includes live-in and live-out data for the dependency information.

20. A method of processing instructions comprising:

fetching a trace descriptor;

separating out a dependency descriptor including dependency information for a set of instructions from the trace descriptor;

fetching the set of instructions described by dependency descriptor; and

executing a plurality of the instruction sequences according to the dependencies stored in the dependency descriptor.

21. A method according to claim 20 further comprising:

updating live-out data in a first storage area.

22. A method according to claim 21 further comprising:

storing the dependency descriptor extracted by the control flow logic into a second storage area; and

reading the descriptor out of the second storage area into the data flow logic.

23. A method according to claim 22 wherein the fetching of a set of instructions is completed just in time for execution.

24. A method according to claim 23 wherein the instructions are out of order.

25. A method according to claim 24 further comprising:

updating the architectural state using the data in the first storage area.

26. A method according to claim 25 further comprising:

recovering an earlier architectural state after a misprediction using the data in the first storage area.

27. A method according to claim 20 further wherein the selecting involves predicting the next trace descriptor to process.

28. A machine-readable medium that provides instructions, which when executed by a machine cause the machine to perform operations comprising:

fetching a trace descriptor;

separating out a dependency descriptor including dependency information for a set of instructions from the trace descriptor;

fetching the set of instructions described by dependency descriptor; and

executing a plurality of the instruction sequences according to the dependencies stored in the dependency descriptor.

29. The machine-readable medium of Claim 28, wherein the operations further comprise:

updating live-out data in a first storage area.

30. The machine-readable medium of Claim 29, wherein the operation further comprise:

storing the dependency descriptor extracted by the control flow logic into a second storage area; and

reading the descriptor out of the second storage area into the data flow logic.